

Toward an Interoperability Reference Model

Rex Buddenberg
Naval Postgraduate School
budden@nps.navy.mil

Abstract

Every discussion of interoperability tends to require an enormous preamble having to do with finding the right layer of a nonexistent reference model. Are we talking about cognitive, doctrinal, data element standardization, networking ...? Or are we talking about elements of information technology that, at best, handle interoperability as a side effect (software portability is an example)? And when we get to prescriptive issues (architecture) are we talking about interoperability between systems or requirements analysis within a single system?

The ISO Reference Model is universally used within the Internet community as a means of organizing the discourse. The Reference Model is properly described as a taxonomy. A means for organizing the discussion.

This paper proposes an Interoperability Reference Model that is intended to perform the same function for interoperable information systems as the ISO Reference Model does for interoperable networks -- organize the discussion.

1. Introduction: Definitions

These are the definitions to accompany Figure 1.

7. Doctrinal interoperability
6. Cognitive or shared situational awareness
5. Interoperable procedures
4. Shared processes
3. Data Element interoperability
2. Information system modularity
1. Internetworkability

Figure 1. A modest proposal

Doctrinal interoperability is a human factor that leads to coherency and uniformity of action. Different decision makers, when presented with the same information will be making similar decisions. The usual doctrinal tensions of uniformity versus creativity are still present and certainly not resolved by this Model. The Model only serves to illustrate the level of abstraction where such discussion belongs.

It may be useful to think about doctrinal interoperability by inversion: we frequently legislate non- interoperability for doctrinal reasons. Use of SIGINT is one case; another is the very careful separation of functions of FBI and CIA.

Doctrinal thinkers will tend to divide this layer into tactical, operational and strategic sublayers.

Cognitive interoperability has to do with shared situation awareness. Information systems are interoperable at this layer if decision makers in two different systems are seeing coherent pictures of the information presented.

The practitioners of interoperability here will point out the truism that the battlefield or a disaster area is very chaotic so you have lots of independent decision makers. The idea is to have them making coherent decisions; 'commander's intent' is part of the lexicon.

Interoperable procedures. This is the domain long inhabited by Standard Operating Procedures (SOP) or Tactics, Techniques & Procedures (TT&P).

Shared processes. This is a software engineering concept. At its trivial level reusable code obviously enhances interoperability but that is a side effect of what is essentially an economy effort in code production. At a more mature level, mobile code and portable code are the pertinent issues. Discussions around acronyms like SOA or SaaS or ERP tend to fit into this layer (although they bleed over into others).

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 21 MAY 2008		2. REPORT TYPE N/A		3. DATES COVERED -	
4. TITLE AND SUBTITLE Toward an Interoperability Reference Model				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release, distribution unlimited					
13. SUPPLEMENTARY NOTES AFCEA-GMU C4I Center Symposium "Critical Issues In C4I" 20-21 May 2008, George Mason University, Fairfax, Virginia Campus					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 4	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

Data. Data element interoperability is a clear requisite to information system interoperability. This is the abstract layer where this discussions of data, meta-data and meta-meta-data belongs. Again, by inversion, it should be clear that two information systems can only be interoperable to the extent that they agree on their data dictionaries.

Modularity. Refers to development of a complex product (or process) from smaller subsystems that can be designed independently. Like the term 'interoperability' modularity is something we all want, but we seem to have little idea how to get it. This is an outlier term in Fig 1 because I can find no constituency for it: nobody in the organization says 'I do modularity as my job'. But if one observes information systems, those that are not interoperable with each other invariably have poor modularity at root.

If we want reusable components, this is clearly the right place in the Reference Model to put the topic:

1. Decoupling of end systems from the communications is the first, most important modularity step. That allows us to change one without the other.
2. Modularity between end systems is a second step – so we can use a Sense module from one information system to feed data to a Decision one in another. Modularity is highly correlated with life cycle maintainability in an information system. The ability of one information system to be interoperable with another has the same attributes as an information system that can grow, evolve, and shed obsolete components over its own life cycle.

Internetwork. At the bottom, communications interoperability can be defined by ability to internetwork. Heterogeneous communications networks are necessary; the key to interoperability is that they can be concatenated together using routers. In Internet terminology, each discrete communications network is a routable network.

2. Methodology and caveats

Imagine a door-knocking exercise. The standard question to each occupant is 'What does interoperability mean to you?' With the exception of modularity, you can find someone who will give you each of the answers above. For example, if you've been raised in the sea services, the term 'ask the chief'

will be familiar. Bang on the door to the CPO mess and ask one of the chiefs what he means by interoperability? The typical answer will be: 'Sir, if we all trained alike we'd be interoperable'. Which fits into #5, interoperable procedures. This is a correct answer, just not a complete one. The effort here is taxonomic (in the sense of Linnaeus), not architectural. If we skip the taxonomic step then efforts at prescriptive architecture will be built on no foundation.

3. Triage

The top three elements in the Reference Model deal with human factors. The processes and data categories apply to interoperability of end systems. And the bottom (network) subject deals, of course, with the internetwork.

While this taxonomy does not prescribe an architecture, it should give a reader some strong hints regarding modularity and where the module boundaries belong.

4. Justifications

The driving need to achieve interoperability at all of these levels of abstraction is 'large information systems. Large information systems are made up of information systems (sense, decide, and act functions connected with communications) that were conceived at smaller scale and then, in real world use, found that they need to share information -- they need to become a system of systems. We need to interconnect these multiple information systems into large information systems and we cannot do that effectively or efficiently without attention at each of these layers of abstraction. (In software engineering, systems are defined as those that require more than one programmer to realize. I'm using 'large' in the same sense here – a large information system is one that crosses program, platform, service, allied boundaries.)

A close corollary of this systems level interoperability (the first four layers) is the need for human interoperability -- between different ranks, between different services and different allies.

5. Prior effort

Previous efforts at erecting such a layered model

include OSD CIO's effort at Levels of Information System Interoperability (LISI). The exercise was well-intentioned but fell short in two significant areas:

- LISI had a point system that rewarded commonality and assumed that commonality would render interoperability. This is closely related to the trap that assumes that standards compliance yields interoperability -- equally fallacious. The notions of modularity and internetworking were largely absent from the LISI scheme.
- The human factors issues of interoperability were not addressed in the LISI model. Without doing so, the incompleteness makes it a hard stretch to get from the need for interoperable infrastructure to the gain in warfighting coherence.

Levels of Conceptual Interoperability Model [1]. This paper has some appeal -- it attempts to take apart 'interoperability' into some components.

- Scope is limited to data interoperability in Figure 1. There is no treatment of the human factors issues nor of the communications interoperability issues.
- Modularity does not appear in the paper. It's difficult to see how the abstractions reached can lead to industrial revolution-style reusable components.
- Maturity vice taxonomy. The progressive levels in this model are gauging growth rather than modularization. In this respect, the model is similar to the Carnegie-Mellon Software Maturity Model.

A second effort worth mentioning is the Defense Information Infrastructure Common Operating Environment. DII COE has many of the same objectives as this Interoperability Reference Model but its focus is on common software modules -- essentially layer 4 of our Interoperability Reference Model. DII COE influences interoperability at other layers, but as side effects. One of the side effects of shared software processes is data commonality so DII COE has influence on layer 3. Another side effect is influence on standard operating procedures and shared situational awareness simply because all users are looking at the same graphical user interface -- one place where commonality does have an interoperability yield.

The Department of Defense has written up an Architecture Framework which is larded with the term

'interoperability,' but 'modularity' only occurs once in Vol 1. DODAF has two shortcomings, a major and a minor: Major. DODAF is not an architecture (whatever your definition); the 'A' is an adjective. Of the two dozen views, the only reference to modularity is in Systems View 1. Whatever value the other views might have for requirements analysis, information systems modularity is not among them. Minor. There is no requirement for any two programs to have the same modularization model.

6. Observations

Like the ISO Reference Model, this Interoperability Reference Model tends to view interoperability as a peer-layer exercise. Routable networks may be heterogeneous at layers 1 and 2 of the ISO Reference Model but must agree on Internet Protocol at layer 3 to be interoperable. Similarly, heterogeneous information systems can be forced into a modicum of interoperability if they agree on data elements. The means of forcing this interoperability may be sneakernetting floppy disks or installing some gateway between the otherwise disparate systems.

Global Command and Control System, as noted above (DII COE) is a good example of progress in a couple of layers with side effects on others. In the early 1990s there were about two dozen tactical decision support system programs in SPAWAR. While each had specific mission objectives, they all had common elements in the support system (e.g. cartography management and track databasing because they were all putting tracklines on charts) which each program was duplicating. The convergence programmatic umbrella was called Copernicus; the engineering solution was called Unified Build and the objective was unification of this unrelated collection of programs. Mapped onto the Interoperability Reference Model, the main thrust was software portability, which Unified Build achieved. Copernicus did nothing about communications interoperability or modularity but in the process of achieving software portability, it had beneficial side effects. The side effect on data interoperability resulted from common software modules tending to tacitly define data elements the same way. Shared situation awareness is another beneficial side effect because viewers of the GCCS screens were all using the same GUI.

The institutional artifact of GCCS development is the Defense Information Infrastructure Common Operating Environment noted above. DII COE is useful within this context -- it defines a framework for achieving and maintaining software portability. But if you attempt to apply DII COE to other layers of the

proposed Interoperability Reference Model, it ceases to be on target.

Shared procedures are affected as a side effect of shared processes -- if disparate users see the same data with the same presentations, they will tend to gravitate to interoperable procedures due to the 'social engineering' embedded in the shared software. This tends toward cognitive interoperability and certainly has influence on doctrine, but again, they are side effects.

7. Doubts and tentatives

The author is fairly confident of the bottom 5 levels or so of this Interoperability Reference Model. But a healthy skepticism should drive further discussion: not entirely sure the model is complete. Are there elements of information system interoperability that are left out entirely? Are the left-outs simply detail that fits in the existing model or is there a layer missing? (Since I've levied this shortcoming against prior efforts, it's only fitting that we reapply that criterion here.)

- the ordering of the layers, particularly in the top half, is suspect.
- relative importance. Over the years, the Presentation Layer of the ISO Reference Model has tended to disappear from discourse entirely (it's a better fit into the data interoperability layer of the Interoperability Reference Model). And the Session Layer of ISO RM has been wedded to Transport Layer functionality that few feel the need to make a distinction. By contrast, IEEE 802 committee has consistently broken the bottom two layers of the ISO RM into four sublayers to assist its standards-making function. Such perturbations can certainly be expected here.

8. Conclusion

A case of successful industrial age interoperability may be of use. NATO small arms is a good case. It was politically impossible, for domestic industrial base reasons, to standardize on a common rifle and pistol for all NATO countries. But NATO did standardize on ammunition; the standard pistol round is 9mm. The interoperable ammunition solution gained interoperability -- the real need -- without imposing unpalatable commonality requirements. The reader will be getting value out of this information age interoperability model if he or she is able to make the same kinds of distinctions.

Perhaps the best concluding remarks are warnings not to read too much into a reference model. Reference models are intended to organize discussions and are not intended to directly solve interoperability problems. Further a taxonomic tool that helps describe the world of information systems is not always a good architectural tool by which we prescribe the next generation of the world. The ISO Reference Model suffered from this abuse in the form of the Government Open System Interconnect Profile.

8. Next Steps

Validation. The taxonomies described are in the nature of a hypothesis. Case studies to validate or void the hypothesis are in order.

Keep the objective in mind: We wish to create a set of interchangeable parts that can be assembled and reassembled into information systems. This modularization model can properly be called a prescriptive architecture. That step is outside the scope of this paper, but if the taxonomy is sensible, then it lays an appropriate foundation.

Reference

- [1] Tolk, A. and J. Muguira, Paper at IEEE 2003 Fall Simulation Interoperability Workshop, Orlando, FL